

# 機械翻訳の精度を考慮した 構文解析器の自己学習

奈良先端科学技術大学院大学

知能コミュニケーション研究室

森下 睦・赤部 晃一・Graham Neubig・吉野 幸一郎・中村 哲

第223回自然言語処理研究会

2015/09/28

NAIST<sup>®</sup>

# 背景

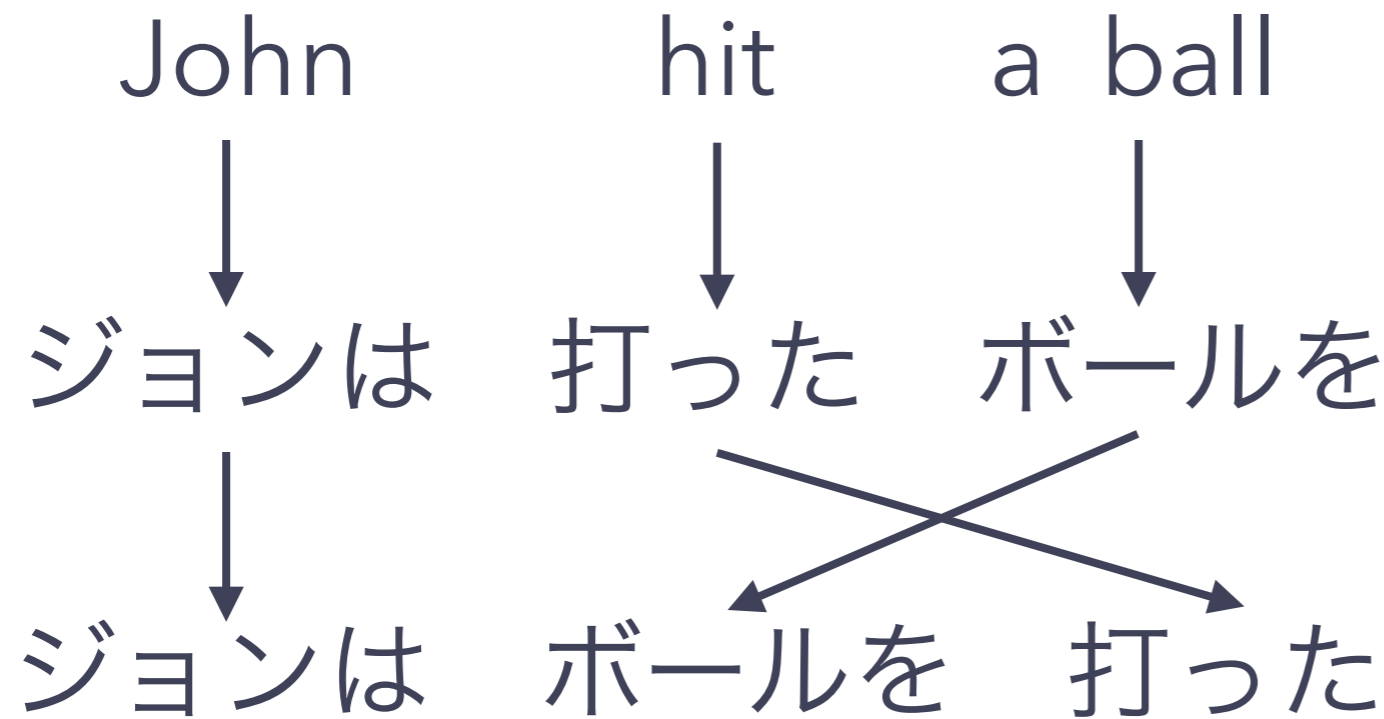
# 統計的機械翻訳



- ◎ 対訳データから翻訳規則を自動的に学習
- ◎ 翻訳規則に基づいて原言語文を目的言語に変換

# フレーズベース翻訳

[Koehn et al., 2003]



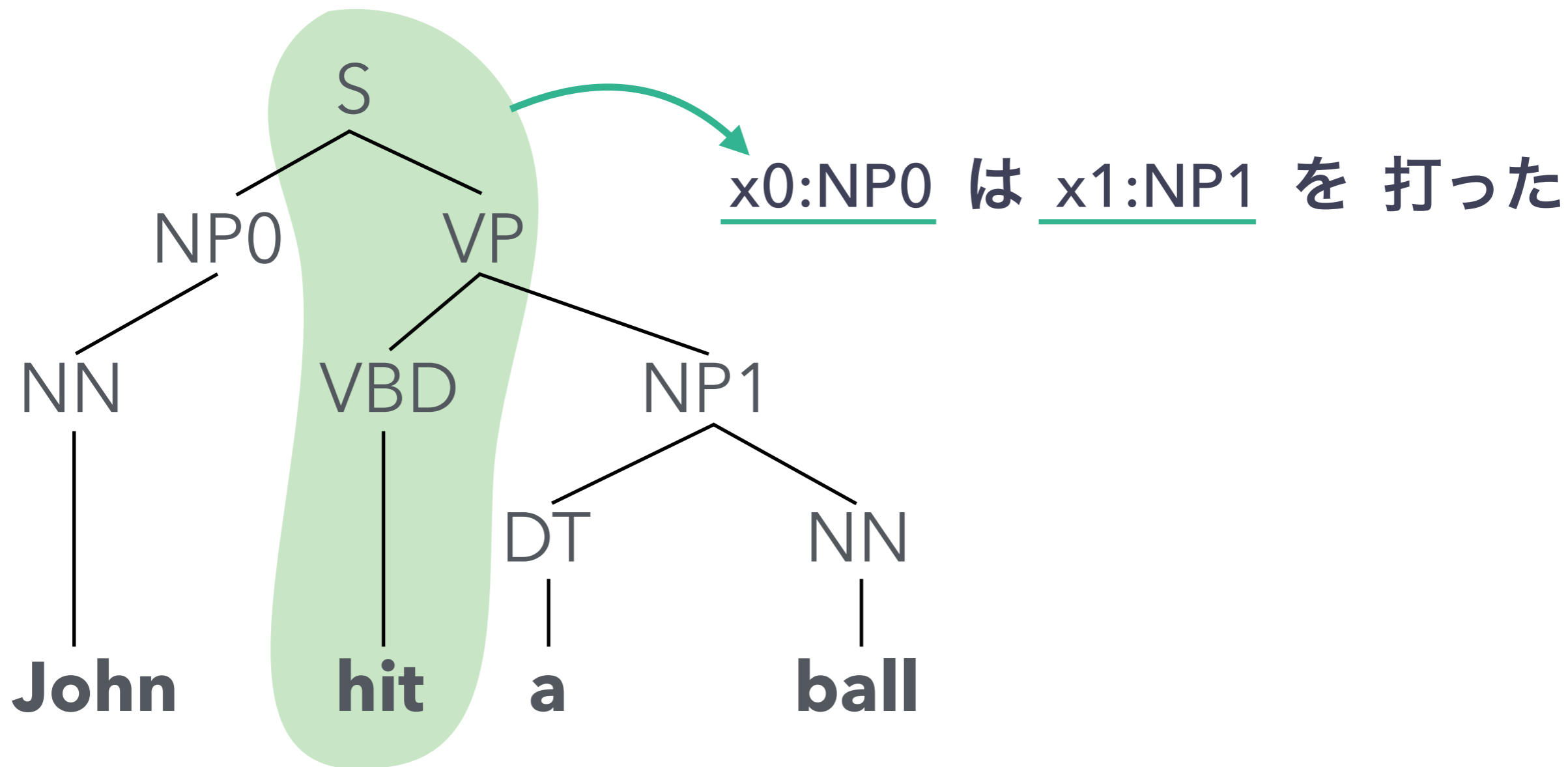
翻訳モデル

並べ替えモデル

- ◎ 文をフレーズごとに翻訳・並べ替え
  - 翻訳モデルの学習が**容易**
  - 語順が大きく異なる言語間での**翻訳精度が低い**

# Tree-to-String 翻訳

[Liu et al., 2006]



- ◎ 原言語の構文木を翻訳に利用
  - 語順が大きく異なる言語間での翻訳精度が高い
  - 構文解析の誤りが翻訳結果に悪影響を及ぼす

# Forest-to-String 翻訳

[Mi et al., 2008]



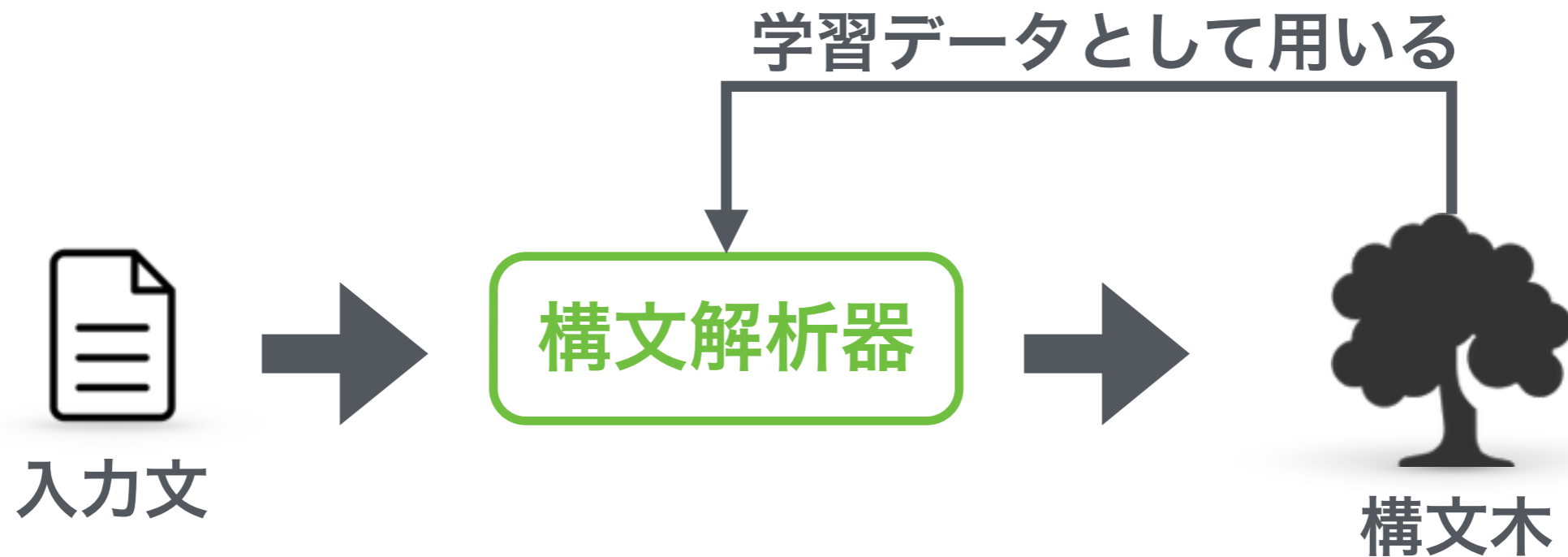
## ◎ 原言語の構文森を翻訳に利用

- 構文木の候補から、翻訳モデルのスコアが高くなる構文木を選択でき、翻訳精度の改善につながる

[Zhang et al., 2012]

# 構文解析器の自己学習

[McClosky et al., 2006]



- ◎ 構文解析器の出力を学習データとする
- ◎ 構文解析器の精度を向上
  - 入力文のドメインへ適応する効果がみられる

# 事前並べ替えにおける 構文解析器の自己学習 [Katz-Brown et al., 2011]

学習データとして用いる



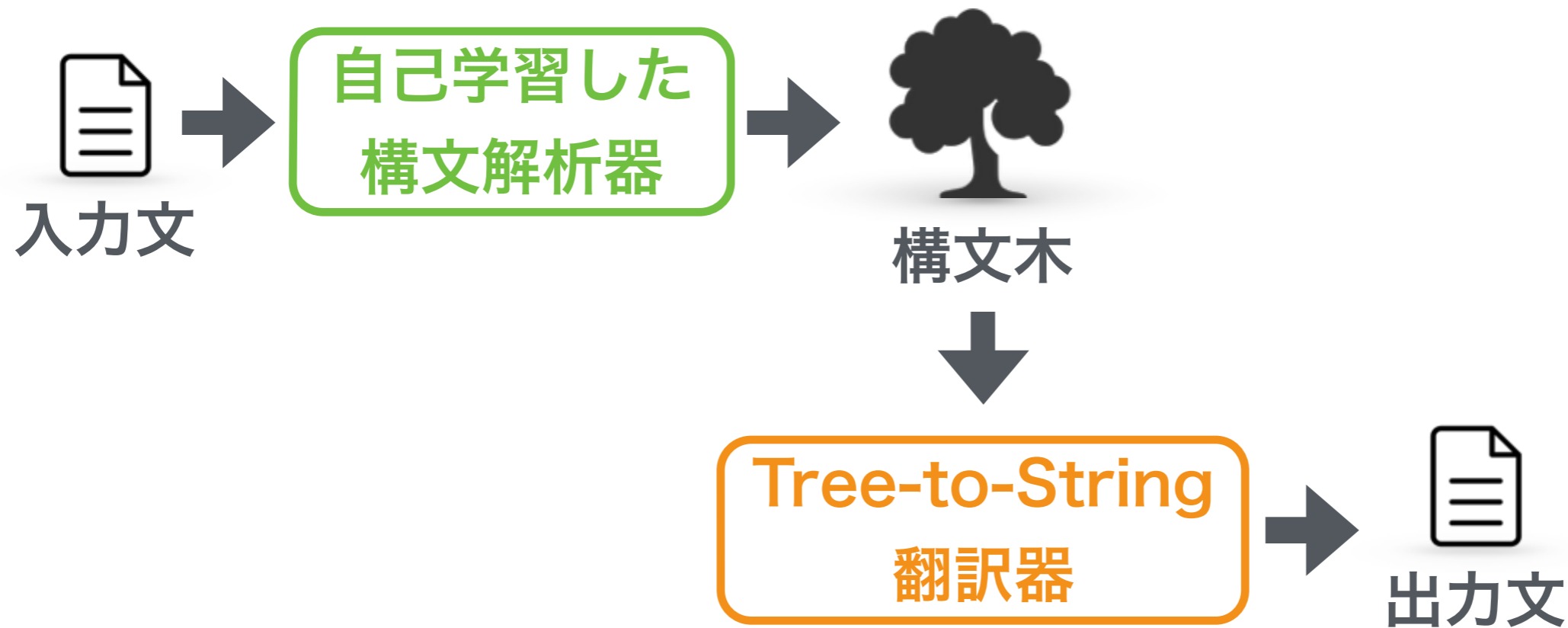
## ◎ 学習するデータを選択する**標的**自己学習による 効果的な自己学習

- 改善に役立つデータのみを学習に用いる
- ただしこの手法では**正解並べ替えデータ**が必要
- 正解データを作るためには**大きなコスト**がかかる



# Tree-to-String 翻訳における 構文解析器の自己学習

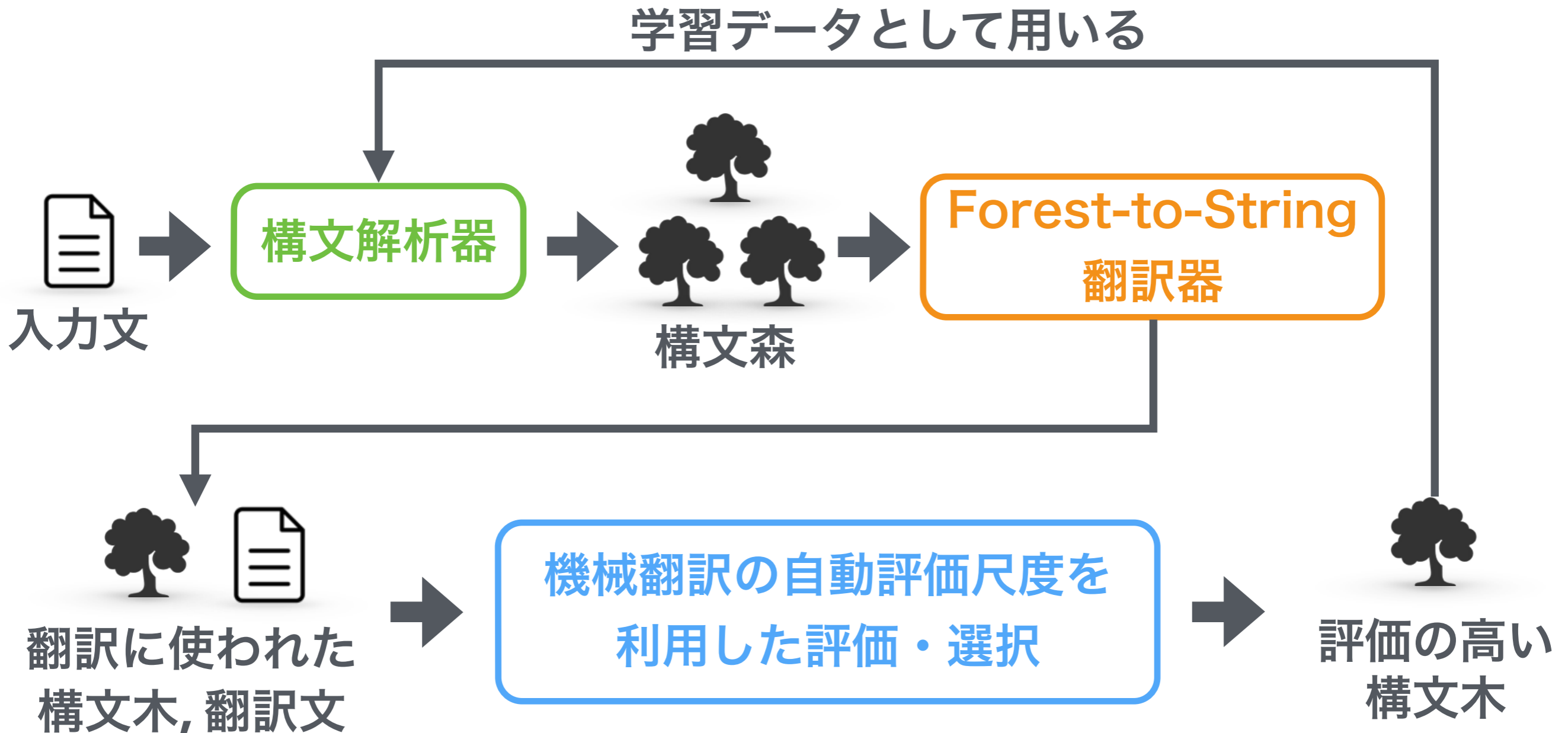
[波多腰 et al., 2015]



- ◎ Tree-to-String 翻訳でも自己学習は**効果あり**
  - 構文解析器の精度が翻訳精度に直接影響するため
  - ただし標的自己学習については**検討されていない**

# 提案手法

# 提案手法



- ◎ **機械翻訳の自動評価尺度を利用した標的自己学習**
  - 低コストかつ正確な評価が可能

# 選択方法

## ◎ 構文木の選択

- 文の構文木の候補からどれを学習するか選択



## ◎ 文の選択

- どの文を自己学習に使用するか選択



# 構文木の選択

## ◎ 解析器 1-best

- 構文解析器が出力した最も確率の高い構文木を使用
- McClosky et al., 2006と同様の手法

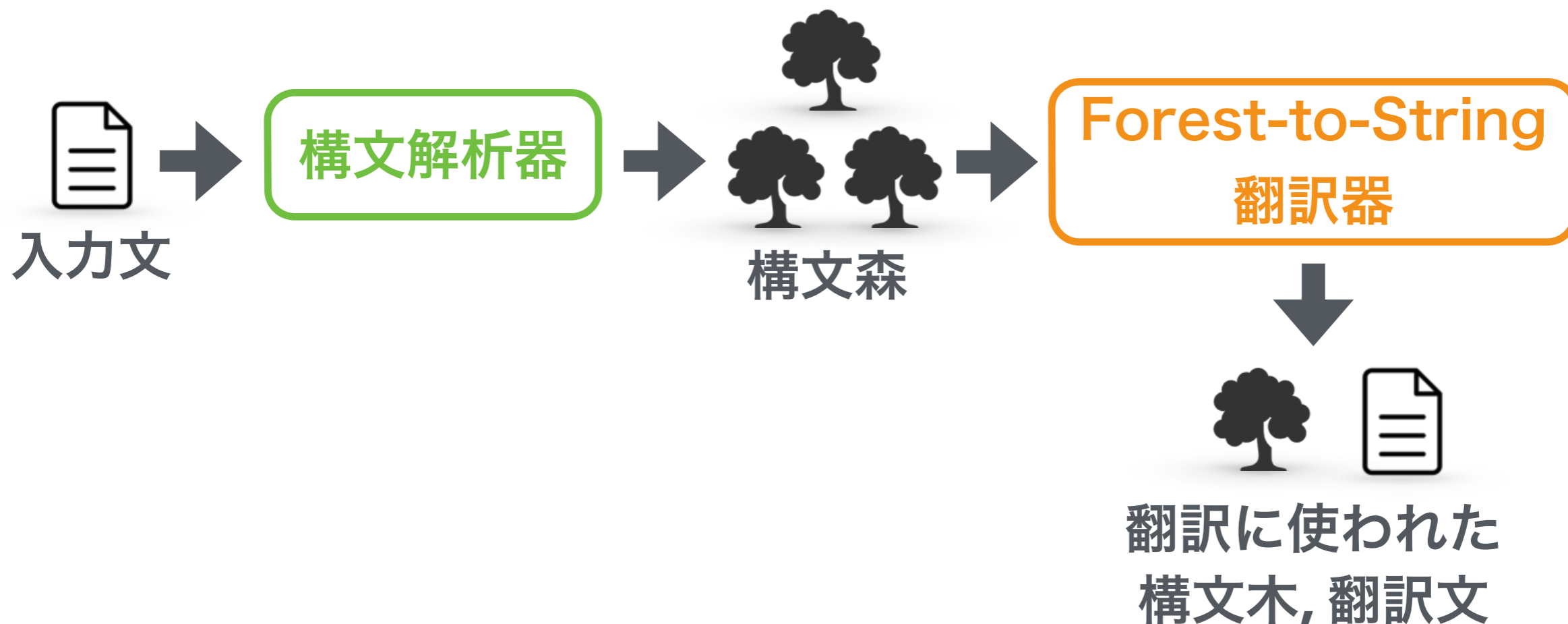
## ◎ 翻訳器 1-best

- 構文森を翻訳し, 訳に使われた構文木を使用

## ◎ 自動評価尺度 1-best

- 訳の候補の中から, 機械翻訳の自動評価値が最も高い訳に使われた構文木を使用
- 自動評価尺度が最も高い訳のことをOracle訳という

# 翻訳器 1-best

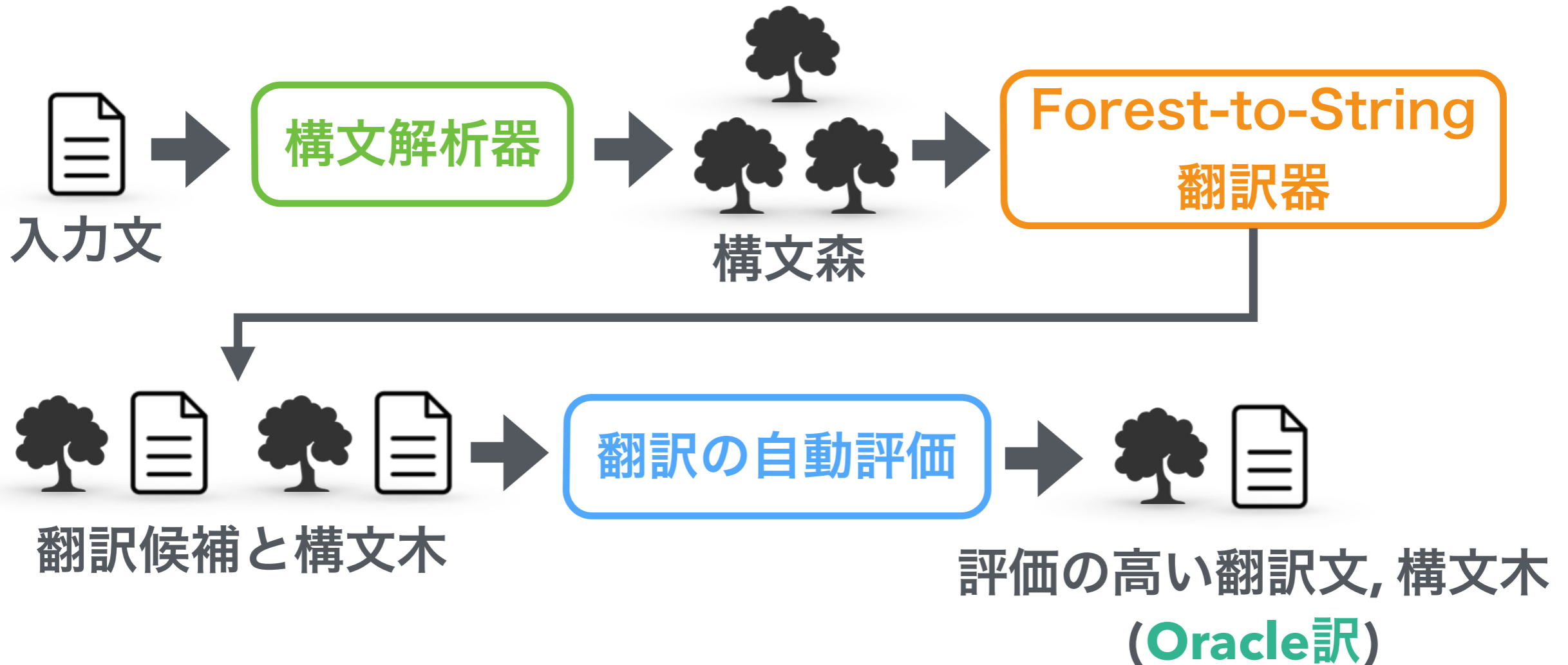


## ● 翻訳器 1-best

- 構文森を翻訳し, 訳に使われた構文木を使用
- 構文森を翻訳した結果, 使われた構文木は精度が高い
- 翻訳器内に正しいと思われる木を選ぶ素性があるため

[Zhang et al., 2012]

# 自動評価尺度 1-best



## ◎ 自動評価尺度 1-best

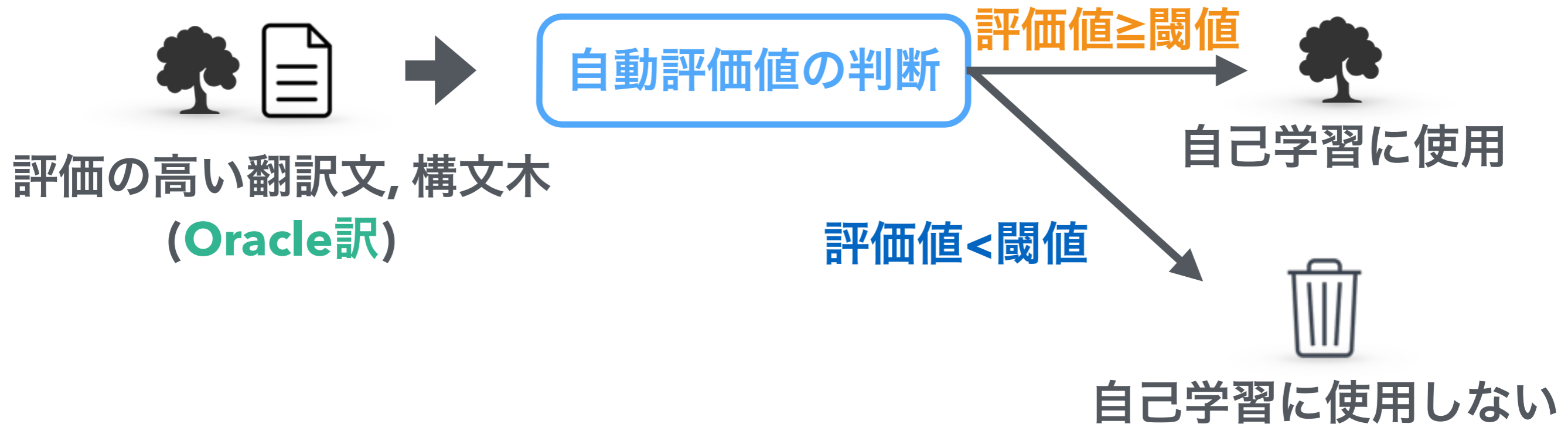
- 訳の候補の中から, 機械翻訳の自動評価値が最も高い訳に使われた構文木を使用
- 自動評価尺度が最も高い訳のことをOracle訳という

# 文の選択

- ◎ **ランダム**
  - 学習候補の文からランダムに文を選択する
- ◎ **自動評価値の閾値**
  - 自動評価値が一定以上の訳の構文木のみ使用
- ◎ **自動評価値の差**
  - 翻訳器 1-bestとOracle訳の自動評価値の差が大きい文の構文木のみを使用



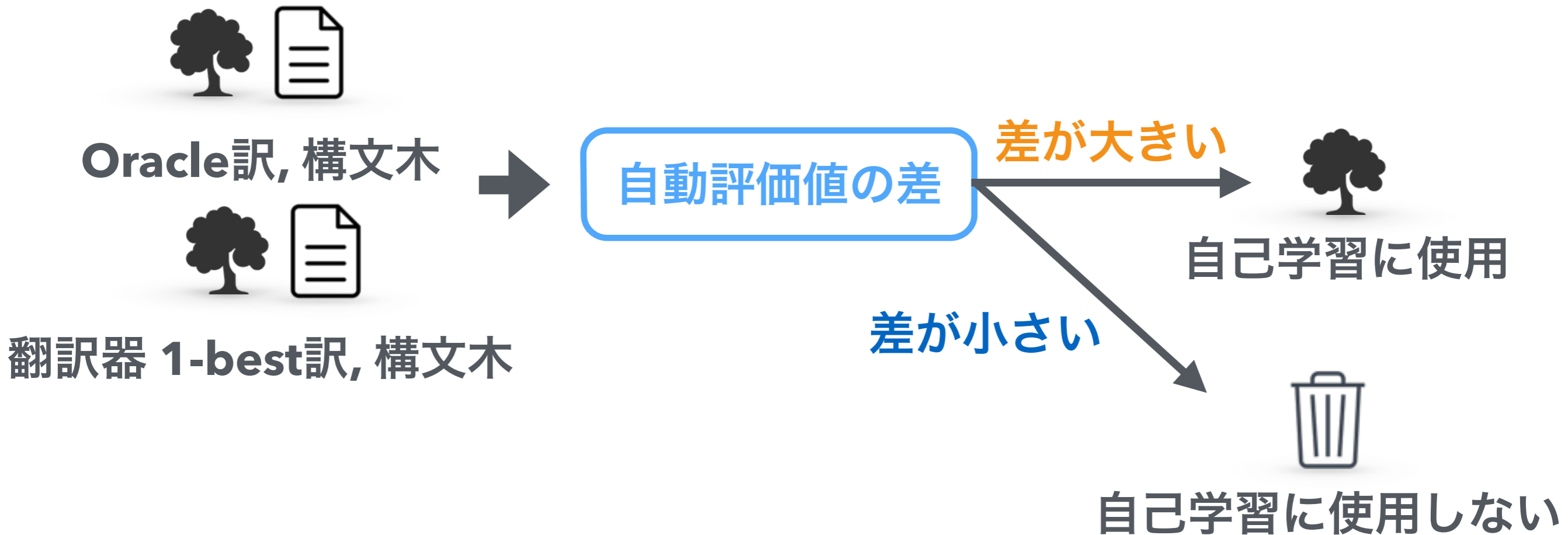
# 自動評価値の閾値



## ◎ 自動評価値の閾値

- 自動評価値が一定以上の訳の構文木のみ使用

# 自動評価値の差



## ◎ 自動評価値の差

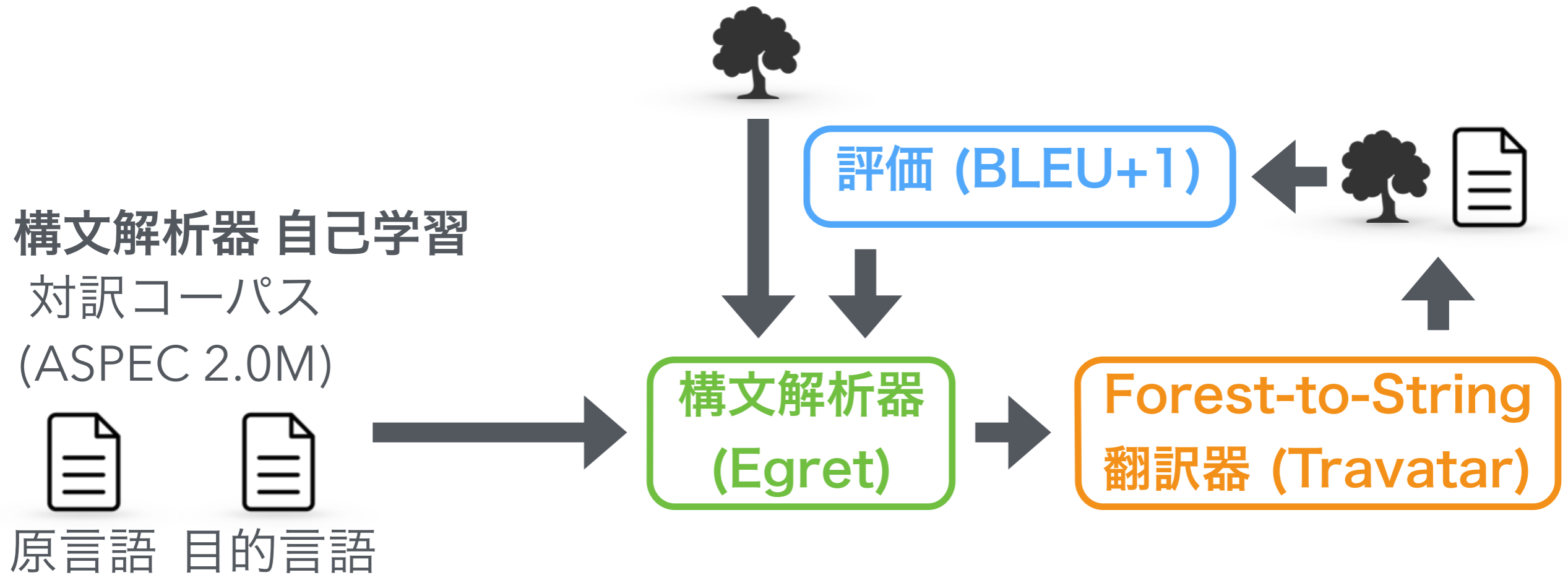
- 翻訳器 1-bestとOracle訳の自動評価値の差が大きい文の構文木のみを使用

# 實驗的評價

# 実験設定 (自己学習時)

既存モデル

日本語係り受けコーパス (7k)



# 実験設定 (精度評価時)

## 翻訳器 学習

対訳コーパス  
(ASPEC 2.0M)



原言語 目的言語



既存の構文解析器  
(Egret)

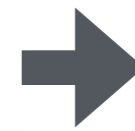


Forest-to-String  
翻訳器 (Travatar)



翻訳文

精度評価



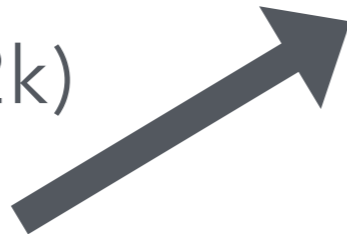
## 翻訳器 開発, テスト

対訳コーパス

(ASPEC dev:2k, test 2k)



原言語 目的言語



自己学習した構文解析器  
(Egret)

日英・日中翻訳を対象

# 実験結果 (日英翻訳)

構文木選択	文選択	Sentences (k)	BLEU	RIBES
<b>Baseline</b>		-	23.83	72.27
解析器 1-best	ランダム	96	23.66	71.77
翻訳器 1-best	ランダム	97	23.81	72.04
Oracle	ランダム	97	23.93	72.09
Oracle	BLEU+1 閾値	120	** 24.26	72.38
Oracle	BLEU+1 差	100	* 24.22	72.32

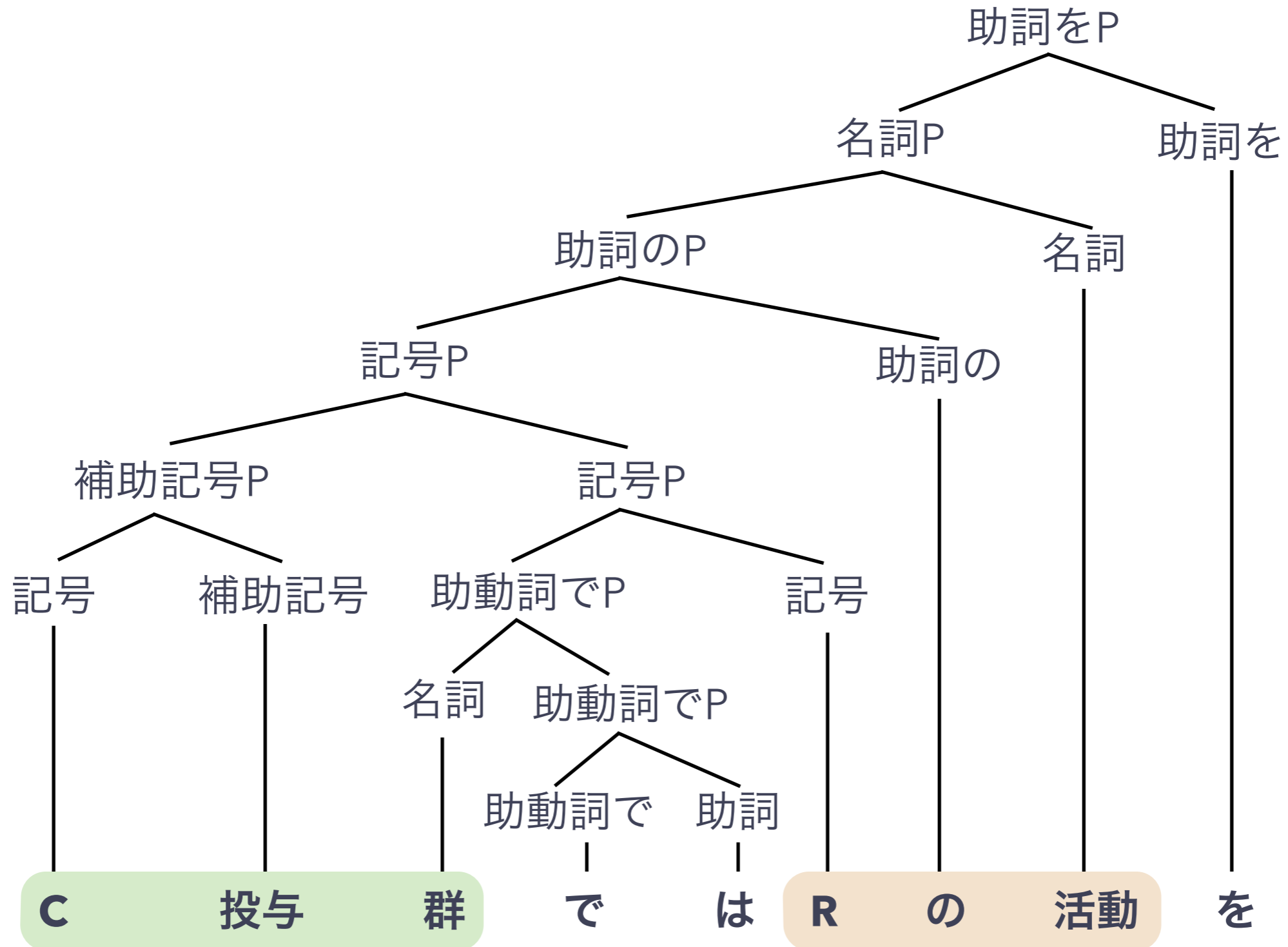
\* :  $p < 0.05$     \*\* :  $p < 0.01$

- ◎ 自己学習により **翻訳精度の有意な改善**

# 自己学習前の誤訳・改善例

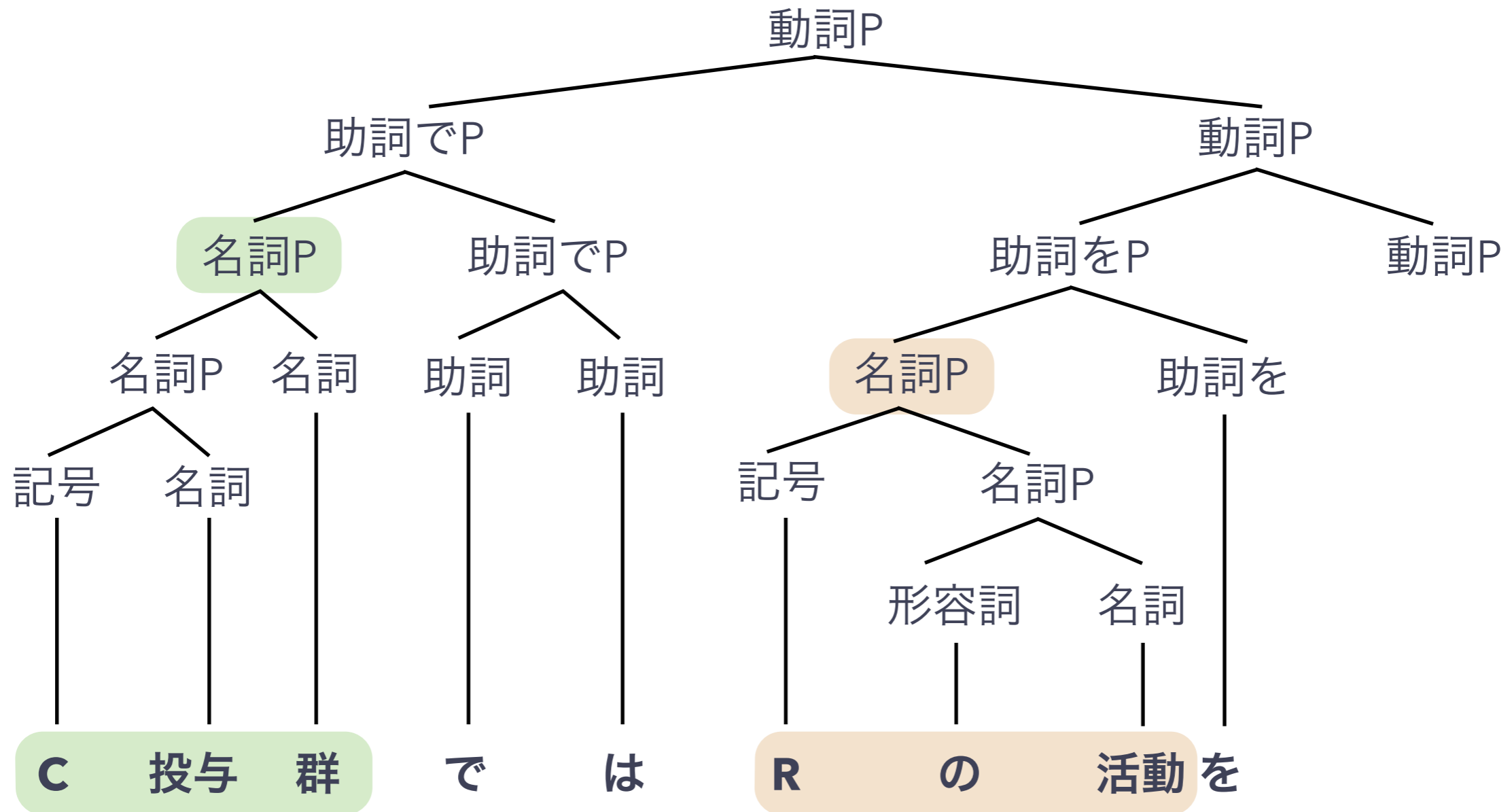
Source	C投与群ではRの活動を240分にかけて明らかに増強した
Reference	in the C - administered group, thermal reaction clearly increased the activity of R for 240 minutes.
Baseline	for 240 minutes clearly enhanced the activity of <u>C administration group R</u> .
Self-Trained	for 240 minutes clearly enhanced the activity of <u>R in the C - administration group</u> .

# 自己学習前の構文木





# 自己学習により改善された構文木



# 実験結果 (日中翻訳)

構文木選択	文選択	Sentences (k)	BLEU	RIBES
Baseline		-	29.60	81.32
解析器 1-best	ランダム	129	29.75	** 81.55
翻訳器 1-best	ランダム	130	29.76	* 81.53
Oracle	ランダム	130	** 29.89	** 81.66
Oracle	BLEU+1 閾値	82	* 29.86	** 81.60
Oracle	BLEU+1 差	100	* 29.85	** 81.59
Oracle (ja-en)	BLEU+1 閾値	120	* 29.87	* 81.58

\* :  $p < 0.05$     \*\* :  $p < 0.01$

- ◎ 自己学習により**翻訳精度の有意な改善**
- ◎ **日英で学習したモデルでも同程度の精度向上**

# 実験結果 (日中翻訳)

構文木選択	文選択	Sentences (k)	BLEU	RIBES
Baseline		-	29.60	81.32
解析器 1-best	ランダム	129	29.75	** 81.55
翻訳器 1-best	ランダム	130	29.76	* 81.53
Oracle	ランダム	130	** 29.89	** 81.66
Oracle	BLEU+1 閾値	82	* 29.86	** 81.60
Oracle	BLEU+1 差	100	* 29.85	** 81.59
Oracle (ja-en)	BLEU+1 閾値	120	* 29.87	* 81.58

\* :  $p < 0.05$     \*\* :  $p < 0.01$

- ◎ 自己学習により**翻訳精度の有意な改善**
- ◎ **日英**で学習したモデルでも同程度の**精度向上**

まとめ

# まとめ

- ◎ **機械翻訳の精度を考慮した  
構文解析器の自己学習により翻訳精度が向上**
  - 日英・日中において有意な向上が見られた
- ◎ **自己学習は目的言語に依存しない可能性**
  - 日英で学習したモデルで日中翻訳の精度向上
- ◎ **今後の課題**
  - さらに多くの言語対で手法が有効であることを確認
  - 目的言語が異なるデータを用いて学習
  - 自己学習を繰り返し行い, 精度が変化するか検証

**END**

# 実験結果 (日英翻訳)

構文木選択	文選択	Sentences (k)	BLEU	RIBES
<b>Baseline</b>		-	23.83	72.27
解析器 1-best	ランダム	96	23.66	71.77
翻訳器 1-best	ランダム	97	23.81	72.04
Oracle	ランダム	97	23.93	72.09
Oracle	<b>BLEU+1 <math>\geq</math> 0.7</b>	206	** 24.27	72.38
Oracle	<b>BLEU+1 <math>\geq</math> 0.8</b>	120	** 24.26	72.38
Oracle	<b>BLEU+1 <math>\geq</math> 0.9</b>	58	** 24.26	72.49
Oracle	<b>BLEU+1 差</b>	100	* 24.22	72.32

\* :  $p < 0.05$     \*\* :  $p < 0.01$

- ◎ 自己学習により **翻訳精度の有意な改善**

# 実験結果 (日中翻訳)

構文木選択	文選択	Sentences (k)	BLEU	RIBES
<b>Baseline</b>		-	29.60	81.32
解析器 1-best	ランダム	129	29.75	** 81.55
翻訳器 1-best	ランダム	130	29.76	* 81.53
Oracle	ランダム	130	** 29.89	** 81.66
Oracle	BLEU+1 $\geq 0.8$	150	** 29.91	81.47
Oracle	BLEU+1 $\geq 0.9$	82	* 29.86	** 81.60
Oracle	BLEU+1 差	100	* 29.85	** 81.59
Oracle (ja-en)	BLEU+1 $\geq 0.8$	120	* 29.87	* 81.58

\* :  $p < 0.05$     \*\* :  $p < 0.01$

- ◎ 自己学習により **翻訳精度の有意な改善**



# なぜ翻訳器1-bestの構文木は 解析器1-bestより良い？

- ◎ Forest-to-String翻訳で考慮される確率
  - 構文木の確率
  - 翻訳モデル
  - 言語モデル
- ◎ 正しい木を使うルールは,  
翻訳ルールの確率が高いため, 選ばれやすい
  - 誤った木に適用できるルールは,  
翻訳確率が低いため, 選ばれにくい
- ◎ 言語モデルにより,  
正しい構文木が選ばれやすい
  - 正しい構文木は言語モデルの確率が高くなりやすい